

## Evy Coding Constructs

**Sequence** step-by-step  
**Selection** `if` statements  
**Iteration** `while` and `for` loops

### Sequence

Declare before use!

```
a:num // declaration
a = 3 // assignment
b := 1 // inferred declaration
print a b // function call
```

### Selection

```
a := rand1
if a < 0.1
  print "below 10%"
else if a <= 0.25 // optional
  print "10 - 25%"
else // optional
  print "above 25%"
end
```

### Iteration

```
// While: condition controlled
i := 1
while i < 8
  print i // 1 3 5 7
  i = i + 2
end

// For: count controlled
// range START STOP STEP
for i := range 1 8 2
  print i // 1 3 5 7
end
```

## Basic Types

```
num // 2, 3.14 default: 0
string // "hi" default: ""
bool // true default: false
```

### Numbers

```
1 + 2 * (6 - 3) // 7, arithmetic
8 % 3 // 2, module (remainder)
3 < 2 // false, less than
3 >= 2 // true, greater or equal
```

### Strings

```
//concatenation
"butter" + "fly" // "butterfly"
// comparison / ordering
"abc" < "efg" // true
```

### Boolean

```
true false // the only bool values
a and b // true if both a, b true
a or b // false if both a, b false
!a // not a, true ↔ false
```

### Comparison

Compare **same types** only, result is **bool**

```
5 == 5 / 2 // equals: false
"a" != "A" // not equals: true

if country == "USA" and age >= 18
  print "adult"
end
```

<https://docs.evy.dev>

## Input / Output Builtins

```
print "a" 4 // a 4
sleep 2 // pause program for 2 sec
s := read // read input into var s
cls // clear printed output
```

### Number Builtins

```
rand1 // random dec num [0,1)
rand 100 // random whole num < 100
max 1 2 // 2: maximum of 1 and 2
pow 2 3 // 8: 2 to the power of 3
```

### Graphic Builtins

```
move x y // move pen to pos x y
circle 2 // radius 2, center pos
rect 1 2 // size 1 x 2, at pos
line x y // from pos to x y

text "a" // draw text at pos
// triangle, poly [x y]... coords
poly [0 0] [60 0] [30 50] [0 0]
```

### Graphic Style Builtins

```
color "blue" // set pen to blue
clear "pink" // fill canvas pink
stroke "red" // outline color
fill "green" // inside color
width 3 // line width, default 0.1

font {size:20 family:"serif"}
// hue 0-360, rest 0-100
hsl hue sat light alpha
hsl 0 100 50 5 // 95% transpt. red
clear (hsl 0 0 0 80) // near black
```